# PENDAHULUAN

## Latar Belakang

## Tujuan

# LANDASAN TEORI

## Studi Pustaka

# METODE PENELITIAN

## Bahan/Data

### Data yang diperoleh

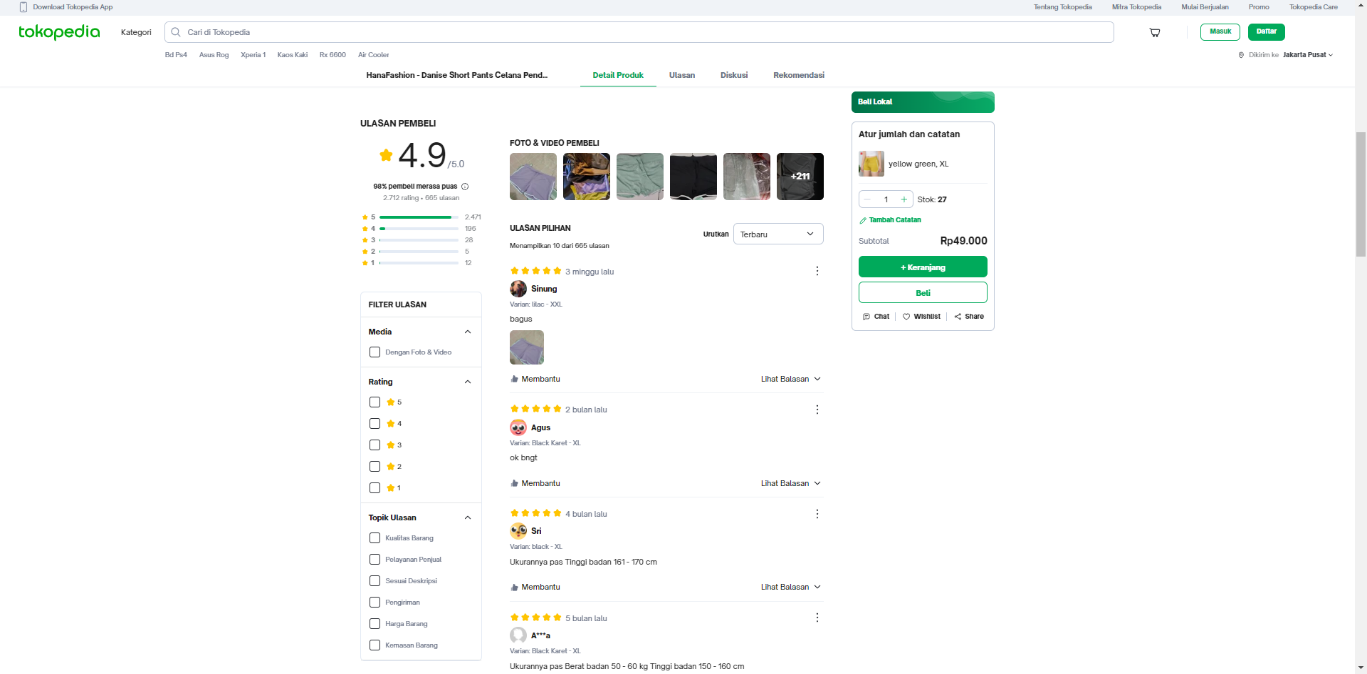
Penelitian ini menggunakan data ulasan toko Hanafashion\_shop yang diambil melalui platform Tokopedia sebagai sumber data primer. Data ulasan pelanggan yang dikumpulkan dari toko Hanafashion\_shop di Tokopedia akan digunakan sebagai dataset utama dalam penelitian ini. Dengan menggunakan data ulasan yang diperoleh secara langsung dari toko Hanafashion\_shop, penelitian ini dapat memberikan wawasan yang lebih spesifik dan relevan terkait dengan pengalaman pelanggan dalam berinteraksi dengan produk-produk yang ditawarkan oleh toko tersebut. Contoh data dapat dilihat pada gambar di bawah.



**Gambar 3. 1** Data Ulasan Tokopedia

### Prosedur Pengumpulan Data

Penelitian ini menggunakan metode scraping menggunakan Selenium untuk mengumpulkan data ulasan toko Hanafashion\_shop di situs web Tokopedia. Metode ini memungkinkan pengendalian browser untuk mengakses halaman toko Hanafashion\_shop, mengambil ulasan pelanggan, dan mengumpulkan data secara otomatis. Dengan demikian, penelitian ini dapat menganalisis ulasan pelanggan dengan efisien dan mendalam untuk memperoleh wawasan yang relevan terkait dengan pengalaman pelanggan di toko Hanafashion\_shop di Tokopedia. Tampilan Ketika melakukan scraping data dapat dilihat pada gambar di bawah.



**Gambar 3. 2** Produk Hanafashion\_shop

Selanjutnya, Selenium akan berinteraksi dengan halaman toko Hanafashion\_shop di situs web Tokopedia untuk mencari ulasan yang relevan. Proses ini mungkin melibatkan mengklik pada halaman produk yang tepat, menavigasi melalui halaman ulasan pelanggan, dan mengumpulkan data ulasan yang ada. Selenium akan mengekstrak teks ulasan dari elemen HTML yang sesuai dan menyimpannya dalam format csv untuk preprocessing lebih lanjut. Langkah-langkah tambahan seperti pembersihan data, penghapusan karakter khusus, atau penggabungan ulasan yang terpisah juga dapat dilakukan untuk memastikan data siap untuk analisis sentimen.

## Alur Penelitian

Tahapan penelitian adalah proses yang dilakukan selama penelitian yang diawali dengan pengumpulan data dengan melakukan scraping hingga penerapan *IndoBert* dan *Support Vector Machine*. Adapun urutan tahapan penelitian dapat dilihat pada gambar di bawah.



**Gambar 3. 3** Alur Penelitian

Data ulasan pelanggan toko Hanafashion\_shop dikumpulkan dari platform Tokopedia dengan menggunakan teknik scraping menggunakan Selenium. Proses pengumpulan data ini mencakup 10 produk yang memiliki jumlah ulasan terbanyak di toko Hanafashion\_shop. Selanjutnya, ulasan-ulasannya digabungkan untuk menghasilkan kumpulan data dari 10 produk berbeda yang dijual oleh toko tersebut. Langkah berikutnya adalah tahap preprocessing, di mana data ulasan dibersihkan dengan menghapus karakter khusus, tautan, dan kata-kata yang tidak relevan dalam bahasa Indonesia. Setelah tahap ini, data dieksplorasi melalui visualisasi untuk memahami perbandingan nilai sentimen dari ulasan pelanggan. Pada tahap analisis sentimen, model IndoBert dan *Support Vector Machine* digunakan untuk mengklasifikasikan ulasan menjadi kategori sentimen positif atau negatif. Selain itu, teknik topic modeling dengan *Hierarchical Dirichlet Process* (HDP) diterapkan untuk mengidentifikasi topik-topik utama dalam ulasan pelanggan. Proses selanjutnya meliputi pelatihan dan pengujian model menggunakan data train dan data test dengan model IndoBert dan *Support Vector Machine*. Performa model dievaluasi menggunakan matriks kebingungan *(confusion matrix)* untuk memahami akurasi dan ketepatan prediksi. Tahap akhir adalah pengujian model dengan memasukkan data ulasan baru untuk mengklasifikasikan sentimen ulasan tersebut sebagai positif atau negatif. Pendekatan ini memberikan wawasan mendalam tentang persepsi pelanggan terhadap toko Hanafashion\_shop di Tokopedia dan mendukung pengambilan keputusan bisnis yang lebih baik.

## IndoBert

Model word embedding yang telah dilatih sebelumnya, dikenal sebagai model pre-trained word embedding, dirancang untuk meningkatkan pemahaman makna dan sintaksis dari teks. Model-model ini dilatih menggunakan dataset besar yang beragam agar dapat mengenali pola bahasa secara luas. Pada tahun 2018, model canggih bernama Bidirectional Encoder Representations from Transformers (BERT) diperkenalkan dan berhasil mencapai hasil unggul dalam berbagai penelitian di bidang pemrosesan bahasa alami (NLP). BERT memanfaatkan arsitektur Transformer dengan mekanisme self-attention untuk memahami konteks hubungan antara kata-kata dalam teks. Di Indonesia, perkembangan signifikan terjadi pada tahun 2020 dengan hadirnya model pre-trained BERT yang dikenal sebagai IndoBERT. Model ini khusus disesuaikan untuk bahasa Indonesia, memungkinkan pemahaman yang lebih baik terhadap bahasa lokal dan meningkatkan performa model dalam tugas-tugas NLP berbahasa Indonesia [1].

Model ini bertujuan untuk meminimalkan gabungan fungsi kerugian dari Masked LM dan Next Sentence Prediction, sehingga menghasilkan model bahasa yang kuat dengan kemampuan yang ditingkatkan dalam memahami konteks dalam kalimat dan hubungan antar kalimat. Ada beberapa hal pada arsitektur BERT, diantaranya :

* BERT BASE memiliki 1 2 lapisan di tumpukan Encoder sedangkan BERT LARGE memiliki 24 lapisan di tumpukan Encoder . Ini lebih dari arsitektur Transformer yang dijelaskan dalam makalah asli ( 6 lapisan encoder ).
* Arsitektur BERT (BASE dan LARGE) juga memiliki jaringan feedforward yang lebih besar (masing-masing 768 dan 1024 unit tersembunyi), dan lebih banyak perhatian (masing-masing 12 dan 16) daripada arsitektur Transformer yang disarankan dalam makalah asli. Ini berisi 512 unit tersembunyi dan 8 kepala perhatian.
* BERT BASE berisi 110 juta parameter sedangkan BERT LARGE memiliki 340 juta parameter.

A close-up of a white rectangular object

Description automatically generated

**Gambar 3. 4** Arsitektur Bert Base & Bert Large

Model ini memproses masukan dimulai dengan token CLS, yang merupakan token klasifikasi, diikuti oleh serangkaian kata sebagai masukan. Token CLS ini berfungsi sebagai penanda awal untuk pemrosesan. Masukan kemudian diteruskan ke lapisan-lapisan di atasnya. Setiap lapisan menggunakan mekanisme perhatian mandiri dan hasilnya kemudian diteruskan ke jaringan feedforward sebelum akhirnya diserahkan ke pembangun enkode berikutnya. Model ini menghasilkan vektor dengan ukuran tersembunyi (768 untuk BERT BASE). Jika kita ingin menggunakan model ini untuk klasifikasi, kita dapat menggunakan keluaran yang terkait dengan token CLS.

A white rectangular object with a black background

Description automatically generated

**Gambar 3. 5** Bert Embeddings

## Support Vector Machine

Support Vector Machine (SVM) adalah teknik yang digunakan untuk melakukan prediksi, baik pada kasus klasifikasi maupun regresi. Prinsip dasar SVM adalah pemisah linier, yaitu kemampuan untuk mengklasifikasikan data yang secara linier dapat dipisahkan. Namun, SVM telah dikembangkan untuk bekerja pada masalah non-linier dengan mengadopsi konsep kernel, memungkinkan model bekerja dalam ruang berdimensi tinggi [2].

### Support Vector Machine untuk Klasifikasi

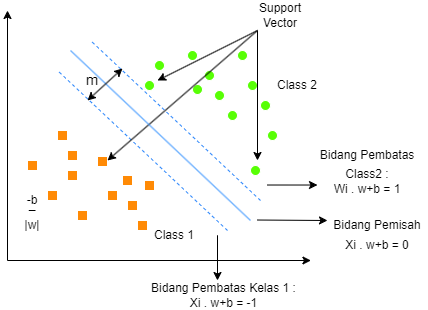
Pada Gambar 3.6, dijelaskan konsep dasar algoritma Support Vector Machine (SVM), yang berfokus pada mencari hyperplane terbaik sebagai pemisah antara dua kelas dalam sebuah data.

*A diagram of a graph

Description automatically generated with medium confidence*

**Gambar 3. 6** Klasifikasi Linear SVM

Pada input space terdapat dua kelas yang berbeda, +1 dan -1, beserta masing-masing pattern yang digambarkan dengan simbol kotak warna orange untuk pattern -1 dan simbol lingkaran hijau untuk pattern +1.



**Gambar 3. 7** Detail Klasifikasi Linear SVM

­Pada gambar 3.7 dijelaskan dalam mengklasifikasi untuk mendapat hasil yang baik hyperplane digunakan untuk memisahkan menjadi dua kelas dengan mengukur margin *hyperplane* tersebut dan mencari titik maksimalnya. Margin adalah jarak antara hyperplane terdekat dengan pattern terdekat dari masing-masing kelas dan pattern yang paling dekat dengan *hyperplane* disebut *support vector*. Seperti gambar dibawah garis tidak putus-putus yangterletak tepat di tengah-tengah kedua keas. Sedangkan *support vector* tampak sebagai *pattern* yang berpotongan dengan garis putus-putus. Dari Gambar 3.6 bidang pemisah dapat dirumuskan :

m = jarak antara dua bidang

w = bidang normal

b = posisi relative terhadap origin

jarak garis dirumuskan wx+b=c ke origin adalah (c-b)/|w|

Margin m dimaksimalkan dengan memenuhi konstrain 2 bidang pembatas yang sejajar dan data yang ada pada bidang pembatas disebut *support vector.* Bidang pembatas kelas pertama membatasi kelas pertama sedangkan bidang pembatas kelas kedua membatasi kelas kedua. Sehingga diperoleh :

*i.w* + *b*  ≥ + 1 *for yi* = +1

*i.w* + *b*  ≥ + 1 *for yi* = -1

Nilai maksimal margin harus memenuhi rumus di atas dan nilai b dan w dikalikan dengan sebuah konstanta yang akan menghasilkan nilai margin yang dikalikan dengan konstanta yang sama. Konstrain merupakan scaling constraint dengan dipenuhi rescaling b dan w. Karena maksimalkan dan minimalkan w dirumuskan dengan pertidaksamaan rumus di atas.

*yi* ( *i.w* + *b*) - 1 ≥ 0

Dengan mengalikan b dan sebuah konstanta, maka menghasilkan nilai m kemudian dikalikan dengan konstanta yang sama. Konstrain merupakan scaling constraint yang dipenuhi dengan rescaling b dan w. Maksimalkan minimumkan .

Untuk mencari nilai margin terbesar untuk bidak pemisah terbaik dapat dirumuskan menjadi masalah optimasi konstrain, yaitu :

*s.t yi(* ≥ 0

Dengan lebih mudah untuk menyelesaikan permasalah optimasi konstrain dalam formulasinya dirubah kedalam formula lagrangian yang menggunakan lagrange multiplier yang diubah menjadi :

Formula pencarian bidang pemisah terbaik ini adalah permasalahan *quadratic programming,* sehingga nilai maksimum global dari akan selalu dapat ditemukan setelah solusi permasalahan *quadratic programming* ditemukan (nilai ), maka kelas dari data pengujian x dapat ditentukan berdasarkan nilai dari fungsi keputusan :

= *support vector,*

Ns = jumlah *support vector*

*=* data yang akan diklasifikasikan